

AMENDMENTS

IN THE CLAIMS:

Add the following claims:

1. (Canceled) A computer readable medium containing computer executable instructions to perform a method for assisting a computer programmer in real time to complete a programming language statement in a computer program, said method comprising:

enabling a programming language editor having a character position cursor and a randomly positionable pointer;

partially compiling available ones of a plurality of programming language statements in said computer program;

defining a finite set of programming language statement information that is relevant to at least one segment of a present programming language statement from among said plurality of programming language statements that is proximate to said character position cursor; and

generating a passive assist window that contains said finite set of programming language statement information in a location proximate to said character position cursor.

2. (Canceled) A method according to claim 1 including:
automatically attempting said steps of claim 1 for each character received by said programming language editor.

3. (Canceled) A method according to claim 2 including:
attempting said steps of claim 1 on a randomly selected one of said plurality of programming language statements in response to a real time request by said computer programmer.

4. (Canceled) A method according to claim 1 including:
attempting said steps of claim 1 on a randomly selected one of said plurality of

programming language statements in response to a real time request by said computer programmer at a time when at least one automatic assist window feature is disabled.

5. (Canceled) A method according claim 1 including:

generating a simultaneous plurality of passive assist windows that each contain a finite set of programming language statement information in a location proximate to said character position cursor, said simultaneous plurality of passive assist windows being selected from at least one of a group comprised of: a selection menu assist window and an informational display assist window.

6. (Canceled) A method according to claim 1 wherein said step of defining includes:

generating said finite list as a selectable list of menu items that can each validly complete said at least one segment of said present programming language statement that is proximate to said character positioned cursor.

7. (Canceled) A method according to claim 1 wherein said step of generating includes:

creating a selection menu assist window comprised of said list of menu items; and enabling window control features for said selection menu assist window.

8. (Canceled) A method according to claim 6 including:

replacing said at least one segment of said present programming language statement with one said list of menu items in response to an input command by said computer programmer.

9. (Canceled) A method according to claim 1 wherein said step of defining includes:

generating an argument list of each argument in said present programming language statement; and

identifying an argument type for each argument in said argument list selected from at least one of a group comprised of: a mandatory argument and an optional argument.

10. (Canceled) A method according to claim 9 wherein said step of generating includes:

reverse parsing said present programming language statement into a plurality of tokens that each represent an individual component selected from at least one of a group comprised of: an object entity segment and a delimiter, in response to a real time request by said computer programmer

distinguishing said plurality of tokens between a procedure call token and any argument token in said argument list; and

binding said argument list.

11. (Canceled) A method according to claim 1 wherein said step of generating includes:

generating an informational display assist window based on an argument list;

201 distinguishing a mandatory argument from an optional argument within said argument list; and

highlighting a present argument within said argument list that corresponds to a present location of said character position cursor within said present programming language statement.

12. (Canceled) A system for passively assisting a user in real time to complete a programming language statement, said system comprising:

a programming language editor having a character position cursor and a randomly positioned pointer;

means for partially compiling available ones of a plurality of programming language statements in said computer program; and

means for generating an assist window that contains a finite set of programming language statement information in a location proximate to said character position cursor, said assist window being selected from at least one of a group comprised of: a selection menu assist window and an informational display assist window.

13. (Canceled) A system according to claim 12 wherein said means for generating includes:

means for identifying a desired menu item from said selection menu assist window; and
means for replacing a segment of a present programming language statement at a present location of said character positioned cursor with said desired menu item in response to said means for identifying.

14. (Canceled) A system according to claim 12 including:
means for displaying information in an informational display assist window, said information being related to at least one segment of a present programming language statement that is proximate a present location of said character position cursor and selected from at least one of a group comprised of : a symbol definition, a defined constant, a procedure call map, and an enumerated list.

15. (Canceled) A system according to claim 12 including:
means for automatically enabling said means of claim 12 for each character received by said programming language editor.

16. (Canceled) A system according to claim 12 including:
means for enabling said means of claim 1 on a randomly selected one of said plurality of programming language statements in response to a real time request by said user and independent of any automatic assist feature.

17. (Canceled) A real time method for assisting a user to complete a programming language statement in a computer program, said real time method comprising:
enabling a programming language editor having a character position cursor;
continuously resolving symbolic portions of available ones of a plurality of programming language statements into a partial program compilation;

identifying a present programming language statement and at least one segment of said present programming language statement based on a location of said character position cursor
determining a finite set of information related to said present programming language statement and said at least one segment of said present programming language statement based on said partial program compilation; and
generating an assist window of said finite set of information.

18. (Canceled) A method according to claim 17 wherein said step of identifying includes:

determining an identity of input to said programming language editor by said user;
enabling a reverse parse evaluation of said present programming language statement into identifiable tokens for each of said at least one segment therein in response to said input being an on-demand request by said user;

enabling execution of a editing task in response to said input being a programming language editor command;

enabling a first type of commit of an identified menu item from a selection menu assist window in response to said input being a commit key, wherein said step of enabling a first type of commit includes;

identifying said commit key as a non-delimiter type commit key; and
discarding said commit key;

enabling a second type of commit of an identified menu item from a selection menu assist window in response to said input being a commit key, wherein said second type of commit includes:

identifying said commit key as a delimiter type commit key; and
inserting said commit key after said identified menu item in said present programming language statement; and

adding to said present programming language statement at a location of said character position cursor in response to said input being a non-commit key type input character.

19. (Canceled) A method according to claim 17 wherein said step of generating includes:

displaying a selection menu assist window where said present programming language statement is identified as an operator embedded programming language statement; and

displaying in informational display assist window where said present programming language statement is identified as a non-operator embedded programming language statement.

20. (Canceled) A method according to claim 19 wherein said non-operator embedded programming language is procedure call.

21. (Original) A computer-readable medium containing computer-executable instructions to perform a method for assisting a computer programmer in real-time to modify a present programming language statement of a computer program, the method comprising:

enabling a programming language editor having a character position cursor and a randomly positionable pointer;

partially compiling available ones of a plurality of programming language statements in said computer program;

defining a finite set of programming language statement information that is relevant to at least one segment of the present programming language statement from among said plurality of programming language statements that is proximate to said character position which allows modification of the programming language statement; and

automatically generating a passive assist window that contains said finite set of programming language statement information in a location proximate to said character position cursor that does not obstruct the current view of said programming language statement.

22. (Original) The computer-readable medium of claim 21, having further computer-executable instructions for performing:

modifying the present programming language statement based at least in part on the selected programming language statement information; and
automatically removing the passive assist window when the programming language statement has been amended.

23. (Original) A computer-readable medium whose contents cause a computer system to supplement a computer programming statement by performing the steps of:

automatically displaying the computer programming statement; and
proximate to the display of an incomplete computer programming statement, automatically displaying a dynamic list of one or more textual programmatic entities.

24. (Original) The computer-readable medium of claim 23, having further computer-executable instructions for performing:

receiving a user input selection of one of the displayed textual programmatic entities; and
adding the selected textual programmatic entity to the displayed statement.

25. (Previously Presented) A method for assisting a computer programmer in modifying a programming language statement, the method comprising:

enabling a positionable cursor within a programming language editing tool;
automatically generating a passive assist window in response to the location of the positionable cursor, the passive assist window containing programming language statement information including at least one of selection menu information and information related to the programming language statement.

26. (Previously Presented) The method of claim 25 wherein the passive assist window generates and disappears independent of user intervention.

27. (Previously Presented) The method of claim 25 wherein the passive assist window permits a user typing the programming language statement to type through the programming language statement information, the passive assist window automatically altering the programming language statement information in response to the typing.

28. (Previously Presented) The method of claim 25 wherein the passive assist window further provides one or more entries for completing the programming language statement via automatic anticipation typing.

29. (Previously Presented) The method of claim 28 wherein the one or more entries are user defined entries.

30. (Previously Presented) The method of claim 29 wherein the user-defined entries enable a plurality of users to modify related programming language statements independent of prior sharing of information related to the user-defined entries.

31. (Previously Presented) The method of claim 25 wherein the passive assist window is a pop-up window that disappears if ignored by a user.

32. (Previously Presented) A computer readable medium containing computer executable instructions to perform a method for assisting a computer programmer in modifying a programming language statement, the method comprising:

enabling a positionable cursor within a programming language editing tool; and
automatically generating a passive assist window in response to a location of the positionable cursor, the passive assist window containing programming language statement information including at least one of selection menu information and information related to the programming language statement.

33 (Previously Presented) The method of claim 32 wherein the passive assist window generates and disappears independent of user intervention.

34. (Previously Presented) The method of claim 32 wherein the passive assist window permits a user typing the programming language statement to type through the programming language statement information, the passive assist window automatically altering the programming language statement information in response to the typing.

21 35. (New) A method for assisting a computer programmer in modifying a programming language statement, the method comprising:

enabling a positionable cursor within a programming language editing tool; and
automatically generating a passive assist window in response to the location of the positionable cursor, the passive assist window containing programming language statement information including at least one of selection menu information and information related to the programming language statement, the selection menu information and information related to the programming language statement being active only when highlighted to enable the computer programmer to ignore the passive assist window.
